

# Customer Onboarding & Workflow Automation Platform

## 1. Project Overview

The Customer Onboarding & Workflow Automation Platform is an enterprise-style application designed to digitize and automate customer onboarding processes. The platform integrates **Appian (workflow and UI)** with a **Spring Boot backend** and an **Oracle database** to handle validations, approvals, and status tracking in a secure, scalable, and auditable manner.

This project mirrors real-world onboarding systems used in banking, fintech, and enterprise IT environments.

---

## 2. Objectives

- Digitize manual customer onboarding processes
  - Implement role-based approval workflows
  - Enforce backend validations and business rules
  - Track onboarding status end-to-end
  - Maintain audit logs for compliance
  - Demonstrate enterprise system design and integration skills
- 

## 3. Scope

### In Scope

- Customer onboarding form
- Backend REST APIs for validation and persistence
- Role-based approval workflow
- Status tracking and audit history
- Oracle database integration
- Appian-Spring Boot REST integration

### Out of Scope (Phase 1)

- External KYC provider integrations
  - Production-grade security hardening
  - Mobile application
-

## 4. System Architecture

### High-Level Architecture

- **Frontend / Workflow Layer:** Appian
- **Backend Layer:** Spring Boot (Java)
- **Database:** Oracle

### Data Flow

1. User submits onboarding request via Appian form
  2. Appian calls Spring Boot REST API
  3. Spring Boot validates and persists data in Oracle
  4. Workflow routes approval tasks based on role
  5. Status updates are synchronized between systems
- 

## 5. Technology Stack

Layer	Technology
Workflow & UI	Appian
Backend	Java, Spring Boot, REST APIs
Database	Oracle
ORM	Spring Data JPA
API Testing	Postman
Documentation	Swagger (OpenAPI)

---

## 6. Functional Requirements

### 6.1 Customer Onboarding

- Capture customer details (name, email, phone)
- Validate mandatory fields
- Prevent duplicate customer creation

### 6.2 Approval Workflow

- Support multi-level approvals (Ops, Manager, Compliance)
- Allow approve/reject with comments
- Enforce role-based actions

## 6.3 Status Tracking

- Track onboarding lifecycle statuses:
- NEW
- IN\_REVIEW
- APPROVED
- REJECTED
- Display real-time status in Appian dashboard

## 6.4 Audit & Compliance

- Store approval history
  - Log approver, role, action, timestamp
- 

# 7. Non-Functional Requirements

- Modular and maintainable codebase
  - Clear separation of concerns
  - Scalable workflow design
  - API response time under acceptable limits
  - Data consistency between Appian and backend
- 

# 8. Database Design

## 8.1 Customer Onboarding Table

- Customer ID (Primary Key)
- Full Name
- Email
- Phone
- Onboarding Status
- Created Timestamp
- Updated Timestamp

## 8.2 Approval History Table

- Approval ID (Primary Key)
  - Customer ID (Foreign Key)
  - Role
  - Action (APPROVE/REJECT)
  - Comments
  - Action Timestamp
-

## 9. API Design

### Core APIs

- **POST /api/onboarding** – Submit onboarding request
- **GET /api/onboarding/{id}** – Fetch onboarding status
- **POST /api/onboarding/{id}/action** – Approve or reject request

### API Responsibilities

- Validate request payloads
  - Enforce business rules
  - Update onboarding status
  - Persist audit logs
- 

## 10. Appian Workflow Design

### Key Appian Objects

- SAIL Interfaces (Forms)
- Process Models (Workflow orchestration)
- Records (Customer & Approval data)
- Groups (Role-based access)
- REST Connected Systems

### Workflow Steps

1. Customer submission
  2. Operations review
  3. Manager approval
  4. Compliance check
  5. Final status update
- 

## 11. Development Plan & Milestones

### Phase 1 – Backend Foundation

- Create Spring Boot project
- Define entities and repositories
- Implement core APIs

### Phase 2 – Workflow Integration

- Build Appian onboarding form
- Configure REST integrations

- Implement approval process model

### **Phase 3 – Enhancements**

- Add audit logging
  - Improve validations
  - API documentation
- 

## **12. Testing Strategy**

- Unit testing for service layer
  - API testing using Postman
  - Workflow testing in Appian
  - End-to-end onboarding scenario validation
- 

## **13. Deliverables**

- Source code (Spring Boot backend)
  - Database schema scripts
  - Appian process models and interfaces
  - API documentation
  - Demo-ready project walkthrough
- 

## **14. Success Criteria**

- Successful onboarding flow from submission to approval
  - Accurate role-based approvals
  - Real-time status updates
  - Clean audit trail for compliance
- 

## **15. Future Enhancements**

- JWT-based authentication
  - Email notifications
  - Integration with external KYC services
  - Event-driven architecture using Kafka
  - Dockerized deployment
- 

**End of Project Plan Document**